HAL open science

Reconstruction attack through classifier analysis

Sébastien Gambs, Ahmed Gmati, Michel Hurfin

To cite this version:

Sébastien Gambs, Ahmed Gmati, Michel Hurfin. Reconstruction attack through classifier analysis. 26th Conference on Data and Applications Security and Privacy (DBSec), Jul 2012, Paris, France. pp.274-281, 10.1007/978-3-642-31540-4_21 . hal-00736945

HAL Id: hal-00736945

https://inria.hal.science/hal-00736945

Submitted on 8 Jun 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Reconstruction Attack through Classifier Analysis

Sébastien Gambs[1,2], Ahmed Gmati[1], and Michel Hurfin[2]

[1] Université de Rennes 1,
Institut de Recherche en Informatique et Systèmes Aléatoires,
Campus de Beaulieu, Avenue du Général Leclerc, 35042 Rennes Cedex, France
{sebastien.gambs,ahmed.gmati}@irisa.fr
[2] Institut National de Recherche en Informatique et en Automatique,
INRIA Rennes - Bretagne Atlantique, France.
michel.hurfin@inria.fr

Abstract. In this paper, we introduce a novel inference attack that we coin as the reconstruction attack whose objective is to reconstruct a probabilistic version of the original dataset on which a classifier was learnt from the description of this classifier and possibly some auxiliary information. In a nutshell, the reconstruction attack exploits the structure of the classifier in order to derive a probabilistic version of dataset on which this model has been trained. Moreover, we propose a general framework that can be used to assess the success of a reconstruction attack in terms of a novel distance between the reconstructed and original datasets. In case of multiple releases of classifiers, we also give a strategy that can be used to merge the different reconstructed datasets into a single coherent one that is closer to the original dataset than any of the simple reconstructed datasets. Finally, we give an instantiation of this reconstruction attack on a decision tree classifier that was learnt using the algorithm C4.5 and evaluate experimentally its efficiency. The results of this experimentation demonstrate that the proposed attack is able to reconstruct a significant part of the original dataset, thus highlighting the need to develop new learning algorithms whose output is specifically tailored to mitigate the success of this type of attack.

Keywords: Privacy, Data Mining, Inference Attacks, Decision Trees.

1 Introduction

Data mining and Privacy may seem *a priori* to have two antagonist goals: Data Mining is interested in discovering knowledge hidden within the data whereas Privacy seeks to preserve the confidentiality of personal information. The main challenge is to find how to extract useful knowledge while at the same time preserving the privacy of sensitive information. *Privacy-Preserving Data Mining* (PPDM) [14, 1, 3] addresses this challenge through the design of data mining algorithms providing privacy guarantees while still ensuring a good level of utility on the output of the learning algorithm.

In this work, we take a first step in this direction by introducing an inference attack that we coined as the *reconstruction attack*. The main objective of this attack is to reconstruct a probabilistic version of the original dataset on which a classifier was learnt from the description of this classifier and possibly some auxiliary information. We propose a general framework that can be used to assess the success of a reconstruction attack in terms of a novel distance between the reconstructed and original datasets. In case of multiple releases of classifiers, we also give a strategy that can be used to merge the different reconstructed datasets into a single one that is closer to the original dataset than any of the simple reconstructed datasets. Finally, we give an instantiation of this reconstruction attack on a decision tree classifier that was learnt using the algorithm C4.5 and evaluate experimentally its efficiency. The results of this experimentation demonstrate that the proposed attack is able to reconstruct a significant part of the original dataset, thus highlighting the need to develop new learning algorithms whose output is specifically tailored to mitigate the success of this type of attack.

The outline of the paper is as follows. First, in Section 2, we describe the notion decision tree that is necessary to understand our paper and we review related work on inference attacks. In Section 3, we introduce the concept of reconstruction attack together with the framework necessary to analyze and reason on the success of this attack. Afterwards, in Section 4, we describe an instantiation of a reconstruction attack on decision tree classifier and evaluate its efficiency on a real dataset. Finally, we conclude in Section 5 by proposing new avenues of research extending the current work.

2 Background and Related Work

Decision tree. Decision tree is a predictive method widely used in data mining for classification tasks, which describes a dataset in the form of a top-down taxonomy [4]. Usually, the input given to a decision tree induction algorithm is a dataset $\mathcal{D}$ composed of n data points, each described by a set of d attributes $\mathcal{A} = \{A_1, A_2, A_3, \ldots, A_d\}$. One of these attributes is a special attribute A_c, called the *class attribute*. The output of the induction algorithm is a rooted tree in which each node is a test on one (or several) attribute(s) partitioning the dataset into two disjoint subsets (*i.e.*, depending on the result of the test, the walk through the tree continues either by following the right or the left branch if the tree is binary). Moreover in a rooted tree, the root is a node without parent and leaves are nodes without children. The decision tree model outputted by the induction algorithm can be used as a classifier $\mathcal{C}$ for the class attribute A_c that can predict the class attribute of a new data point $x_?$ given the description of its non-class attributes. The construction of a decision tree is usually done in a top-down manner by first setting the root to be a test on the attribute that is the most discriminative according to some splitting criterion that varies across different tree induction algorithms. The path from the root to a leaf is unique and it characterizes a group of individuals at the same time

by the class at the leaf but also by the path followed. In his seminal work, Ross Quinlan has introduced in 1986 an induction tree algorithm called ID3 (*Iterative Dichotomiser 3*) [11]. Subsequently, Quinlan has developed an extension to this algorithm called C4.5 [12], which incorporates several extensions such as the possibility to handle continuous attributes or missing attribute values. However, both C4.5 and ID3 rely on the notion of *information gain*, which is directly based on the Shannon entropy [13], as a splitting criterion.

Inference attacks. An *inference attack* is a data mining process by which an adversary that has access to some public information or the output of some computation depending on the personal information of individuals (plus possibly some auxiliary information) can deduce private information about these individuals that was not explicitly present in the data and that was normally supposed to be protected. In the context of PPDM, a *classification attack* [8] and *regression attack* [9] working on decision trees were proposed by Li and Sarkar. The main objective of these two attacks is to reconstruct the attribute class of some of the individuals that were present in the dataset on which the decision tree has been trained. This can be seen as a special case of the *reconstruction attack* that we propose in this work that aims at reconstructing not only the class attribute of a data point but also the other attributes. It was also shown by Kifer that the knowledge of the data distribution (which is sometimes public) can help the adversary to cause a privacy breach. More precisely, Kifer has introduced the *deFinetti attack* [5] that aims at building a classifier predicting the sensitive attribute corresponding to a set of non-sensitive attributes. Finally, we refer the reader to [7] for a study evaluating the usefulness of some privacy-preserving techniques for preventing inference attacks.

3 Reconstruction Attack

3.1 Reconstruction Problem

In our setting, the adversary can observe a classifier $\mathcal{C}$ that has been computed by running a learning algorithm on the *original dataset* $\mathcal{D}_{orig}$. The main objective of the adversary is to conduct a *reconstruction attack* that reconstruct a *probabilistic version of this dataset*, called $\mathcal{D}_{rec}$, from the description of the classifier $\mathcal{C}$ (and possibly some auxiliary information Aux) that is as close as possible from the original dataset $\mathcal{D}_{orig}$ according to a distance metric Dist that we defined later.

Definition 1 (Probabilistic dataset). *A* probabilistic dataset $\mathcal{D}$ *is composed of* n *data points* $\{x_1, \ldots, x_n\}$ *such that each datapoint* x *corresponds to a set of* d *attributes* $\mathcal{A} = \{A_1, A_2, A_3, \ldots, A_d\}$. *Each attribute* A_k *has a domain of definition* $\mathcal{V}_k$ *that includes all the possible values of this attribute if this attribute is* categorical *or corresponds to an interval* $[min, max]$ *if the attribute is* numerical. *The knowledge about a particular attribute is modeled by a probability distribution over all the possible values of this attribute. If a particular value of the attribute*

*gathers all the probability mass (*i.e.*, its value is perfectly determined), then the attribute is said to be* deterministic*. By extension, a probabilistic dataset whose attributes are all deterministic (*i.e.*, the knowledge about the dataset is perfect) is called a* deterministic dataset*.*

In this work, we assume that the original dataset $\mathcal{D}_{orig}$ is deterministic in the sense that it contains no uncertainty about the value of a particular attribute and no missing values. From this dataset $\mathcal{D}_{orig}$, a classifier $\mathcal{C}$ is learnt and the adversary will reconstruct a probabilistic dataset $\mathcal{D}_{rec}$. For the sake of simplicity in this paper, we also assume that the adversary has no prior knowledge about some attributes being more likely than others. Therefore, if for a particular attribute A_k of a datapoint x, the adversary hesitates between two different possibles values then both values are equally probable for him (*i.e.*, uniform prior). In the same manner, if the adversary knows that the value of a particular attribute belongs to a restricted interval $[a, b]$ then no value within this interval seems more probable to him than other. Finally, in the situation in which the adversary has absolutely no information about the value of a particular attribute, we use the symbol "$*$" to denote this absence of knowledge (*i.e.*, $A_k = *$ if the adversary has no knowledge about the value of the kth, attribute or even $x = *$ if the adversary has no information at all about a particular data point).

3.2 Evaluating the Quality of the Reconstruction

In order to evaluate the quality of the reconstruction, we define a distance between two datasets that quantifies how close these two datasets are. We assume that the two datasets are of same size and that before the computation of this distance they have been *aligned* in the sense that each data point of one dataset has been paired with one (and only one) data point of the other dataset.

Definition 2 (Distance between probabilistic datasets). *Let D and D' be two probabilistic datasets each containing n data points (*i.e.*, respectively $D = \{x_1, \ldots, x_n\}$ and $D' = \{x'_1, \ldots, x'_n\}$) such that each datapoint x corresponds to a set of d attributes $\mathcal{A} = \{A_1, A_2, A_3, \ldots, A_d\}$. The* distance *between these two datasets $\mathsf{Dist}(D_1, D_2)$ is defined as*

$$\mathsf{Dist}(D_1, D_2) = \frac{1}{nd} \sum_{i=1}^{n} \sum_{k=1}^{d} \frac{H(V_k(x'_i) \cup V_k(x_i))}{H(V_k)}, \tag{1}$$

for which $V_k(x'_i) \cup V_k(x_i)$ corresponds to the union of the values for the kth attribute of x_i and x'_i, V_k is all the possible values of this k^{th} attribute (or all the discretized values in case of an interval) and H denotes the Shannon entropy.

Basically, this distance quantifies for each data point and each attribute, the uncertainty that remains about the particular value of an attribute if the two knowledges are pooled together. In particular, this distance is normalized and will be equal to zero *if and only if* it is computed between two copies of the

same deterministic dataset (*e.g.*, $\mathsf{Dist}(\mathcal{D}_{orig},\mathcal{D}_{orig})= 0$). On the other extreme, let D_* be a probabilistic dataset in which the adversary is totally ignorant of *all the attributes of all the data points* (*i.e.*, $\forall k$ such that $1 \leq k \leq d$, $\forall i$ such that $1 \leq i \leq n$, $V_k(x_i) = *$). In this situation, $\mathsf{Dist}(D_*,D_*)= 1$ as the distance simplifies to $\mathsf{Dist}(D_*, D_*) = \frac{1}{nd}\sum_{i=1}^{n}\sum_{k=1}^{d}\frac{H(|V_k|)}{H(|V_k|)} = \frac{nd}{nd}$. For a reconstructed dataset $\mathcal{D}_{rec}$, the computation of the distance between this dataset and itself returns a value between 0 and 1 that quantifies the level of uncertainty (or conversely the amount of information) in this dataset.

While Definition 2 is generic enough to quantify the distance between two probabilistic datasets, in our context we will mainly use it to compute the distance between the probabilistic dataset $\mathcal{D}_{rec}$ and the deterministic dataset $\mathcal{D}_{orig}$. More precisely, we will use the value of $\mathsf{Dist}(\mathcal{D}_{rec},\mathcal{D}_{orig})$ as the *measure of success of a reconstruction attack*.

3.3 Continuous Release of Information

In this work, we are also interested in the situation in which a classifier is released on a regular basis (*i.e.*, not just once), after the additions of new data points to the dataset. We now define the notion of *compatibility* between two probabilistic datasets, which is in a sense also a measure of distance between these two datasets.

Definition 3 (Compatibility between probabilistic datasets). *Let D and D' be two probabilistic datasets each containing n data points (*i.e.*, respectively $D = \{x_1, \ldots, x_n\}$ and $D' = \{x'_1, \ldots, x'_n\}$) such that each datapoint x corresponds to a set of d attributes $\mathcal{A} = \{A_1, A_2, A_3, \ldots, A_d\}$. The* compatibility *between these two datasets $\mathsf{Comp}(D_1, D_2)$ is defined as*

$$\mathsf{Comp}(D_1, D_2) = \frac{1}{nd}\sum_{i=1}^{n}\sum_{k=1}^{d}\frac{H(V_k(x'_i) \cap V_k(x_i))}{H(V_k)}, \tag{2}$$

for which $V_k(x'_i) \cap V_k(x_i)$ corresponds to the intersection of the values for the kth attribute of x_i and x'_i, V_k is all the possible values of this k^{th} attribute (or all the discretized values in case of an interval) and H denotes the Shannon entropy.

Note that the formula of the compatibility between two datasets is the same as for the distance with the exception of using the intersection rather than the union when pooling together two different knowledges about the possible values of the kth attribute of a data point x. The main objective of the compatibility is to measure how much the uncertainty is reduced by combining the two different datasets into one. Suppose for instance, that $\mathcal{D}$ and $\mathcal{D}'$ are respectively the reconstruction obtained by performing a reconstruction attack on two different classifiers $\mathcal{C}$ and $\mathcal{C}'$. .

Merging reconstructed data sets. Let us consider that a first classifier $\mathcal{C}$ has been generated at some point in the past. Later, in the future, new records have been added to the dataset and another classifier $\mathcal{C}'$ is learnt on this updated version of the dataset. We assume that an adversary can observe the two classifiers $\mathcal{C}$ and $\mathcal{C}'$ and apply a reconstruction attack on $\mathcal{C}$ and $\mathcal{C}'$ to build respectively two probabilistic datasets $\mathcal{D}$ and $\mathcal{D}'$. In order to merge these two datasets $\mathcal{D}$ and $\mathcal{D}'$ To merge the two probabilistic datasets $\mathcal{D}$ and $\mathcal{D}'$ into one single probabilistic dataset, denoted $\mathcal{D}_{rec}$, the adversary can adopt the following strategy.

1. Apply the reconstruction attack on the classifiers $\mathcal{C}$ and $\mathcal{C}'$ to obtain respectively the reconstructed datasets $\mathcal{D}$ and $\mathcal{D}'$ (we assume without loss of generality that the size of $\mathcal{D}'$ is smaller or equal to the size of $\mathcal{D}$).
2. Pad $\mathcal{D}'$ with extra data points that have perfect uncertainty (*i.e.*, $x = *$) until the size of $\mathcal{D}'$ is the same as the size of $\mathcal{D}$.
3. Apply the *Hungarian algorithm* [6, 10] in order to align $\mathcal{D}$ and $\mathcal{D}'$. Defining an alignment amounts to sort one of the datasets such that the ith record x_i of $\mathcal{D}$ corresponds to the ith record x'_i of $\mathcal{D}'$. The Hungarian method solves the alignment problem and finds the optimal solution that maximizes the compatibility $\mathsf{Comp}(\mathcal{D}\ ,\ \mathcal{D}')$ between two sets of n data points.
4. Merge $\mathcal{D}$ and $\mathcal{D}'$ into a single reconstructed dataset $\mathcal{D}_{rec}$ by using the alignment computed in the previous step. For each attribute A_k, the domain of definition the merged point is made of the intersection of $V_k(x) \cap V_k(x')$ if this intersection is non-empty and set to the default value $*$ otherwise.
5. Compute the distance metric $\mathsf{Dist}(\mathcal{D}_{rec},\mathcal{D}_{orig})$ for evaluating the success of the reconstruction attack.

4 Reconstruction Attack on Decision Tree

Let $\mathcal{C}$ be a classifier that has been computed by running a C4.5 algorithm on the original dataset $\mathcal{D}_{orig}$. This decision tree classifier is the input of our reconstruction algorithm. For each branch of the tree, the sequence of tests composing this branch form the description of probabilistic data points that will be reconstructed out of this branch. The reconstruction algorithms follows a branch either in a top-down manner and refines progressively the domain of definition $V_k(x)$ for each attribute A_k of a probabilistic data point x until the leaf is reached. As we have run a version of C4.5 in which each leaf also contains the number of data points for each class, we can add the corresponding number of probabilistic data points of each class with the refined description to the probabilistic dataset $\mathcal{D}$ under construction. The algorithm explores all the branches of tree to reconstruct the whole probabilistic dataset $\mathcal{D}$.

To evaluate the success of this reconstruction attack on a decision tree classifier, we have run an experiment on the "Adult" dataset from UCI Machine Learning Repository [2]. This dataset is composed of $d = 14$ attributes such as age or marital status, including the income attribute, which is either "> 50K" or "<= 50" and that we have used as class attribute during the construction of the decision tree. To construct the C4.5 classifiers, we have used the WEKA

software [15]. Moreover, for each attribute A_k, we have computed its domain of definition V_k, which is defined by the finite set of possible values. For continuous attribute such as age, the extremal values observed the complete database were used to determine the minimal and maximal possible values. The experimentations were performed in a random subset of 200 records of the original "Adult" dataset and not on the complete database. We denote this subset of 200 records by $\mathcal{D}_{orig}$ and the reconstruction attack aims at reconstructing this particular dataset. The metric Dist is used to evaluate the success of the reconstruction attack. The smaller this distance, the more accurate the reconstruction is. Figure 1 displays the result of our experiments obtained when computing the distance between a reconstructed learnt on a dataset whose number of points varies between 1 to 200. Not surprisingly, we can see from these results that a reconstruction attack performed on a classifier that contains more information about the original dataset leads to a reconstruction that is more accurate (*i.e.*, closer to the original dataset).

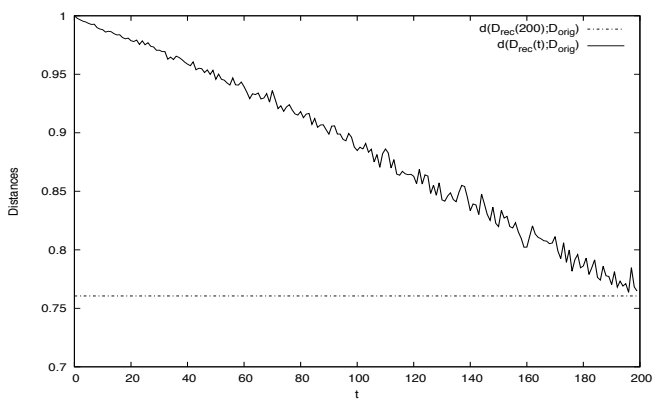

Fig. 1. Distance between a reconstructed dataset D' and $\mathcal{D}_{orig}$ when the reconstruction attack is run on a decision tree learnt on a number of data points varying between 1 and 200 (the size of $\mathcal{D}_{orig}$).

We have also conducted several experiments in which two reconstructed datasets learnt from classifiers released at different time where merged using the algorithm described in Section 3.3. Our main finding is that it is possible to obtain a limited gain in the order of 0.01 or 0.02 when combining the two datasets (we leave the details of these experiments for the full version of the paper due to lack of space).

5 Conclusion

In this paper, we have introduced the concept of reconstruction attack whose aim is to reconstruct a probabilistic version of the original dataset from the descrip-

tion of a classifier. We have also proposed a novel distance based on information entropy that measures the closeness between the original and the reconstructed datasets and can be used to assess the success of the attack. Moreover, we have design a specific instance of a reconstruction attack and demonstrate his efficiency on a real dataset coming from the UCI repository. The current work is only the first step towards the development of a framework for evaluating the impact of releasing a classifier for the privacy of the dataset. As future works, we want to design reconstruction attack for other types of classifiers such as neural networks or ensemble methods such as boosting. We also want to develop a method for merging several reconstructed datasets into a single coherent one in case of multiple releases.

References

1. Aggarwal, C.C., Yu, P.S. (eds.): Privacy-Preserving Data Mining - Models and Algorithms, Advances in Database Systems, vol. 34. Springer (2008)
2. Asuncion, A., Frank, A.: UCI machine learning repository (2010), http://archive.ics.uci.edu/ml
3. Bertino, E., Lin, D., Jiang, W.: A survey of quantification of privacy preserving data mining algorithms. In: Aggarwal and Yu [1], pp. 183–205
4. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley, New York, 2. edn. (2001)
5. Kifer, D.: Attacks on privacy and definetti's theorem. In: Çetintemel, U., Zdonik, S.B., Kossmann, D., Tatbul, N. (eds.) SIGMOD Conf. pp. 127–138. ACM (2009)
6. Kuhn, H.W.: The hungarian method for the assignment problem. Naval Research Logistics Quarterly 2, 83–97 (1955)
7. Li, C., Shirani-Mehr, H., Yang, X.: Protecting individual information against inference attacks in data publishing. In: Ramamohanarao, K., Krishna, P.R., Mohania, M.K., Nantajeewarawat, E. (eds.) DASFAA. Lecture Notes in Computer Science, vol. 4443, pp. 422–433. Springer (2007)
8. Li, X.B., Sarkar, S.: Against classification attacks: A decision tree pruning approach to privacy protection in data mining. Operations Research 57(6), 1496–1509 (2009)
9. Li, X.B., Sarkar, S.: Protecting privacy against regression attacks in predictive data mining. In: Galletta, D.F., Liang, T.P. (eds.) ICIS. pp. 1–15. Association for Information Systems (2011)
10. Munkres, J.: Algorithms for the assignment and transportation problems. Journal of the Society for Industrial and Applied Mathematics 5, 32–38 (March 1957)
11. Quinlan, J.R.: Induction of decision trees. Machine Learning 1(1), 81–106 (1986)
12. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann (1993)
13. Shannon, C.E.: A mathematical theory of communication. The Bell Systems Technical Journal 27, 379–423, 623–656 (1948)
14. Verykios, V.S., Bertino, E., Fovino, I.N., Provenza, L.P., Saygin, Y., Theodoridis, Y.: State-of-the-art in privacy preserving data mining. SIGMOD Record 33(1), 50–57 (2004)
15. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann (1999)